ANDPAD

How resolve Gem dependencies in your code?

Hiroshi SHIBATA @hsbt

2023/09/21 Euruko 2023

Copyright © 2020 Present ANDPAD Inc. This information is confidential and was prepared by ANDPAD Inc. for the use of our client. It is not to be relied on by and 3rd party. Proprietary & Confidential 無断転載・無断複製の禁止

Hiroshi SHIBATA

@hsbt

Ruby core team

RubyGems/Bundler team

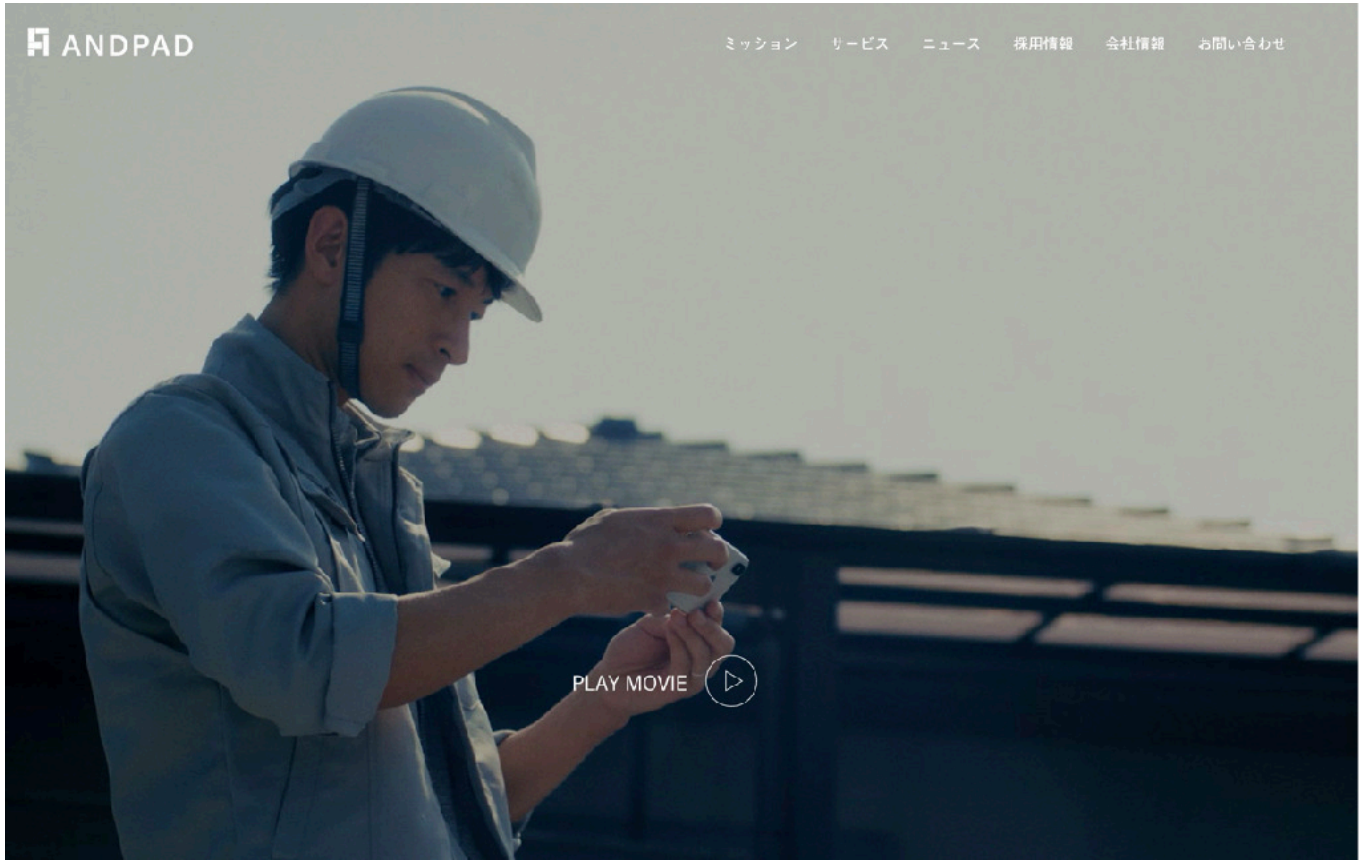
Technical fellow at ANDPAD

https://hsbt.org



Copyright © 2020 Present ANDPAD Inc. This information is confidential and was prepared by ANDPAD Inc. for the use of our client. It is not to be relied on by and 3rd party. Proprietary & Confidential 無断転載・無断複製の禁止

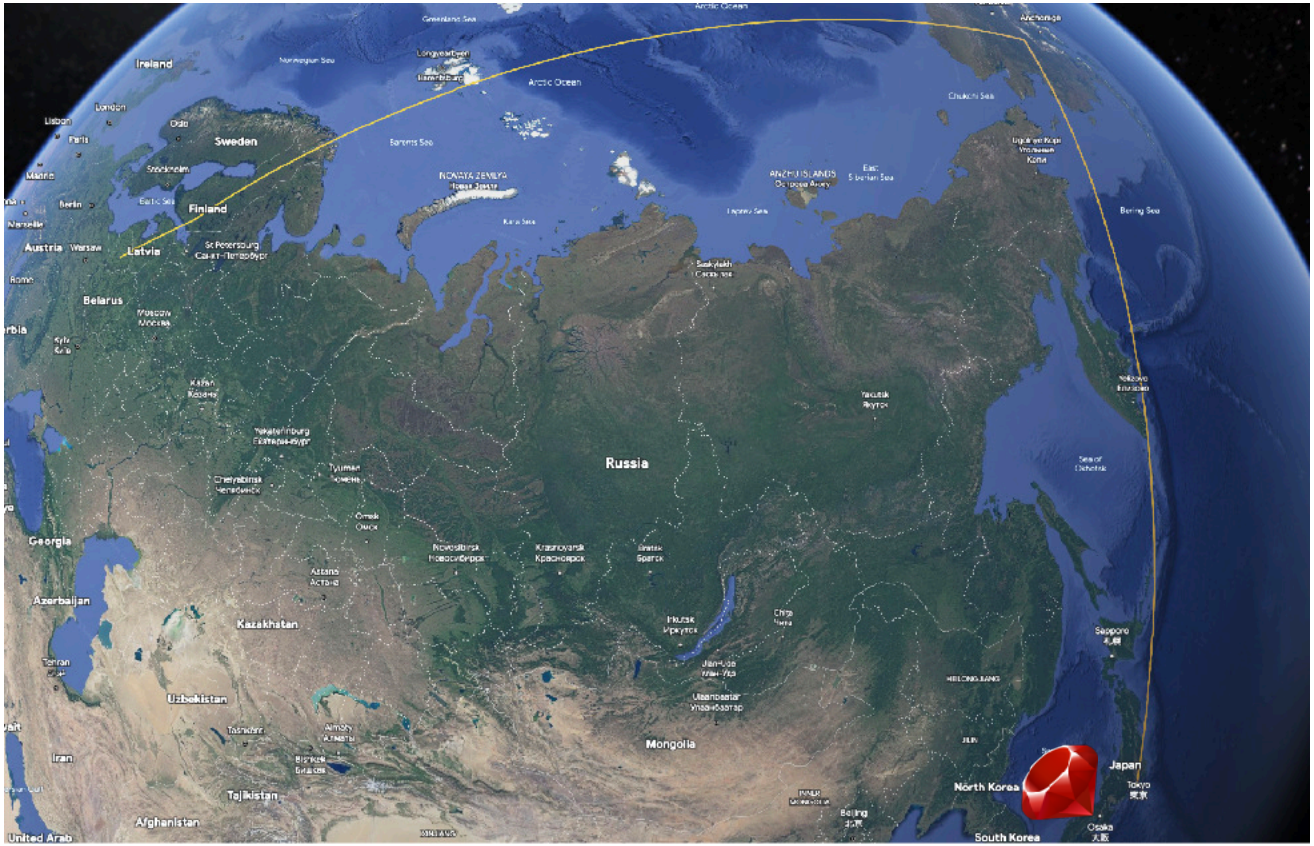
Introduction of ANDPAD



Copyright © 2020 Present ANDPAD Inc. This information is confidential and was prepared by ANDPAD Inc. for the use of our client. It is not to be relied on by and 3rd party.　Proprietary & Confidential　無断転載・無断複製の禁止

I'm from Japan where is Ruby birth place

ANDPAD



Copyright © 2020 Present ANDPAD Inc. This information is confidential and was prepared by ANDPAD Inc. for the use of our client. It is not to be relied on by and 3rd party.　Proprietary & Confidential　無断転載・無断複製の禁止

RubyKaigi 2023 is Ruby conference in Japan





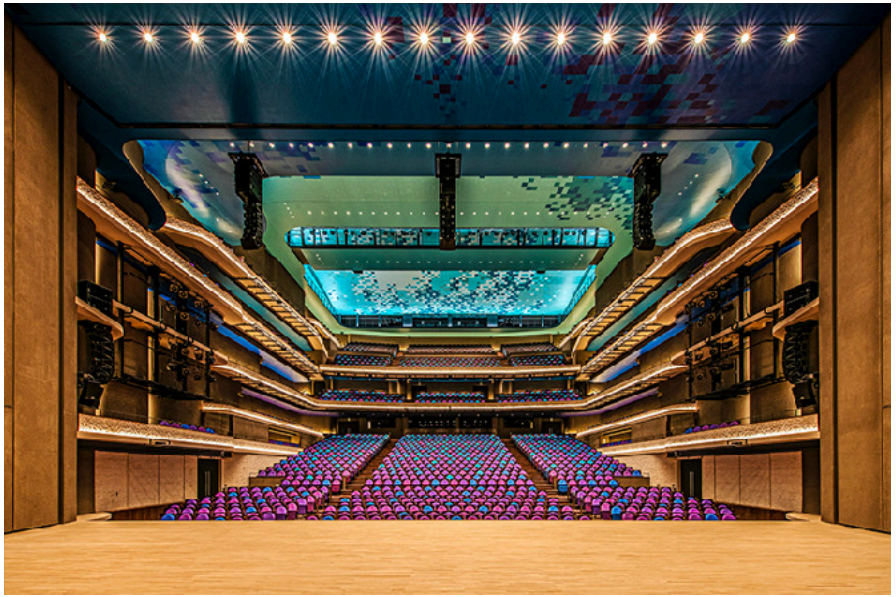




Copyright © 2020 Present ANDPAD Inc. This information is confidential and was prepared by ANDPAD Inc. for the use of our client. It is not to be relied on by and 3rd party. Proprietary & Confidential 無断転載・無断複製の禁止

RubyKaigi 2024 will be coming Okinawa island in Japan at May, 2024.





Copyright © 2020 Present ANDPAD Inc. This information is confidential and was prepared by ANDPAD Inc. for the use of our client. It is not to be relied on by and 3rd party.　Proprietary & Confidential　無断転載・無断複製の禁止

ANDPAD

What's Dependency Resolution with RubyGems/Bundler?

Copyright © 2020 Present ANDPAD Inc. This information is confidential and was prepared by ANDPAD Inc. for the use of our client. It is not to be relied on by and 3rd party. Proprietary & Confidential 無断転載・無断複製の禁止

ANDPAD

What are RubyGems and Bundler?

Copyright © 2020 Present ANDPAD Inc. This information is confidential and was prepared by ANDPAD Inc. for the use of our client. It is not to be relied on by and 3rd party. Proprietary & Confidential 無断転載・無断複製の禁止

What's package manager?

ANDPAD

- User interface
  - Provide commands like install/update/search package
- Dependency Resolution
  - Resolve dependencies of package and provide list of name and version of package
- Version locking (NEW!)
  - Provide environment to lock specified versions of package

Copyright © 2020 Present ANDPAD Inc. This information is confidential and was prepared by ANDPAD Inc. for the use of our client. It is not to be relied on by and 3rd party. Proprietary & Confidential 無断転載・無断複製の禁止

ANDPAD

RubyGems is a package manager of the Ruby language.

- rubygems/rubygems.org:
  - The Ruby community's gem host.
  - rubygems.org is maintain by infrastructure team of rubygems. It is different team from rubygems cli team.
- rubygems/rubygems:
  - Command line tool for rubygems.org
  - Now, rubygems maintained rubygems team. I'm member of this team.

Copyright © 2020 Present ANDPAD Inc. This information is confidential and was prepared by ANDPAD Inc. for the use of our client. It is not to be relied on by and 3rd party. Proprietary & Confidential 無断転載・無断複製の禁止

- Gem
  - A package/library for the Ruby programming language
- Gem::Specification
  - Class for defining metadata including name, version, platform, etc.
- gemspec
  - File describing the Gem::Specification in RubyGems/Bundler
    - This file is written by you for releasing gem
    - This file is created at gem install time by RubyGems

Copyright © 2020 Present ANDPAD Inc. This information is confidential and was prepared by ANDPAD Inc. for the use of our client. It is not to be relied on by and 3rd party.　Proprietary & Confidential　無断転載・無断複製の禁止

What's Bundler?

ANDPAD

- Bundler is also package manager of Ruby language
  - Bundler focused version locking feature
  - Bundler extends a lot of RubyGems resources like gemspec.
- Bundler works with Gemfile like:

```
# frozen_string_literal: true

source "https://rubygems.org"

gemspec

gem "rake", ">= 11.1”
```

Copyright © 2020 Present ANDPAD Inc. This information is confidential and was prepared by ANDPAD Inc. for the use of our client. It is not to be relied on by and 3rd party.　Proprietary & Confidential　無断転載・無断複製の禁止

ANDPAD

Inside of Ruby libraries and gem dependencies.

Copyright © 2020 Present ANDPAD Inc. This information is confidential and was prepared by ANDPAD Inc. for the use of our client. It is not to be relied on by and 3rd party. Proprietary & Confidential 無断転載・無断複製の禁止

ANDPAD

- You can see gemspec with `Gem.loaded_specs` like this:

```
>> Gem.loaded_specs["rack"]
=>
Gem::Specification.new do |s|
  s.name = "rack"
  s.version = Gem::Version.new("2.2.8")
  s.installed_by_version = Gem::Version.new("3.4.10")
  s.authors = ["Leah Neukirchen"]
  s.date = Time.utc(2023, 7, 31)
  s.dependencies = [Gem::Dependency.new("minitest", Gem::Requirement.new(["~> 5.0"]), :development),
   Gem::Dependency.new("minitest-sprint", Gem::Requirement.new([">= 0"]), :development),
   Gem::Dependency.new("minitest-global_expectations", Gem::Requirement.new([">= 0"]), :development),
   Gem::Dependency.new("rake", Gem::Requirement.new([">= 0"]), :development)]
  s.description = "Rack provides a minimal, modular and adaptable interface for developing web
applications in Ruby. By wrapping HTTP requests and responses in\nthe simplest way possible, it unifies
and distills the API for web\nservers, web frameworks, and software in between (the so-
called\nmiddleware) into a single method call.\n"
(...snip...)
end
```

- `Gem::Specification#dependencies` is important parts of your application.

Copyright © 2020 Present ANDPAD Inc. This information is confidential and was prepared by ANDPAD Inc. for the use of our client. It is not to be relied on by and 3rd party. Proprietary & Confidential 無断転載・無断複製の禁止

What's Default gems

ANDPAD

- The Ruby core team released "Default gems" to the rubygems.org.
  - You can install standard libraries of Ruby via RubyGems.
  - Default gems are openssl, psych, json, etc… You can see all of default gems at https://stdgems.org/
- Rubygems have a detection method for default gems.

```
>> require 'rss'
=> true
>> Gem.loaded_specs["rss"].default_gem?
=> false
>> require 'openssl'
=> true
>> Gem.loaded_specs["openssl"].default_gem?
=> true
```

Copyright © 2020 Present ANDPAD Inc. This information is confidential and was prepared by ANDPAD Inc. for the use of our client. It is not to be relied on by and 3rd party. Proprietary & Confidential 無断転載・無断複製の禁止

What's Bundled gems

ANDPAD

- We bundled *.gem and unpacked files to tarball package for Bundled gems with `gems/bundled_gems` in ruby/ruby repository like this:



nobu Update test-unit

8 contributors

8 lines (7 sloc) | 361 Bytes

```
1  did_you_mean 1.3.0 https://github.com/yuki24/did_you_mean
2  minitest 5.11.3 https://github.com/seattlerb/minitest
3  net-telnet 0.2.0 https://github.com/ruby/net-telnet
4  power_assert 1.1.4 https://github.com/k-tsj/power_assert
5  rake 12.3.2 https://github.com/ruby/rake
6  test-unit 3.3.3 https://github.com/test-unit/test-unit
7  xmlrpc 0.3.0 https://github.com/ruby/xmlrpc
```

- `make install` installed Bundled gem your box.

Copyright © 2020 Present ANDPAD Inc. This information is confidential and was prepared by ANDPAD Inc. for the use of our client. It is not to be relied on by and 3rd party. Proprietary & Confidential 無断転載・無断複製の禁止

Performance issues with RubyGems

ANDPAD

- RubyGems extends `require`, `gem` and `warn`.
  - Because Ruby startup time is slow with RubyGems.
- Bundler resolve this slow down.

```
def require(path) # :doc:
  return gem_original_require(path) unless
Gem.discover_gems_on_require

  begin
    RUBYGEMS_ACTIVATION_MONITOR.enter

    path = path.to_path if path.respond_to? :to_path

    if spec = Gem.find_unresolved_default_spec(path)
      # Ensure -I beats a default gem
      resolved_path = begin
        rp = nil
    (snip)
```

Copyright © 2020 Present ANDPAD Inc. This information is confidential and was prepared by ANDPAD Inc. for the use of our client. It is not to be relied on by and 3rd party. Proprietary & Confidential 無断転載・無断複製の禁止

- Bundler also extends RubyGems classes/methods. It's enabled when we used Bundler.
- `Gem::Specification#extension_dir` needs to handle git resource of Gemfile like this:

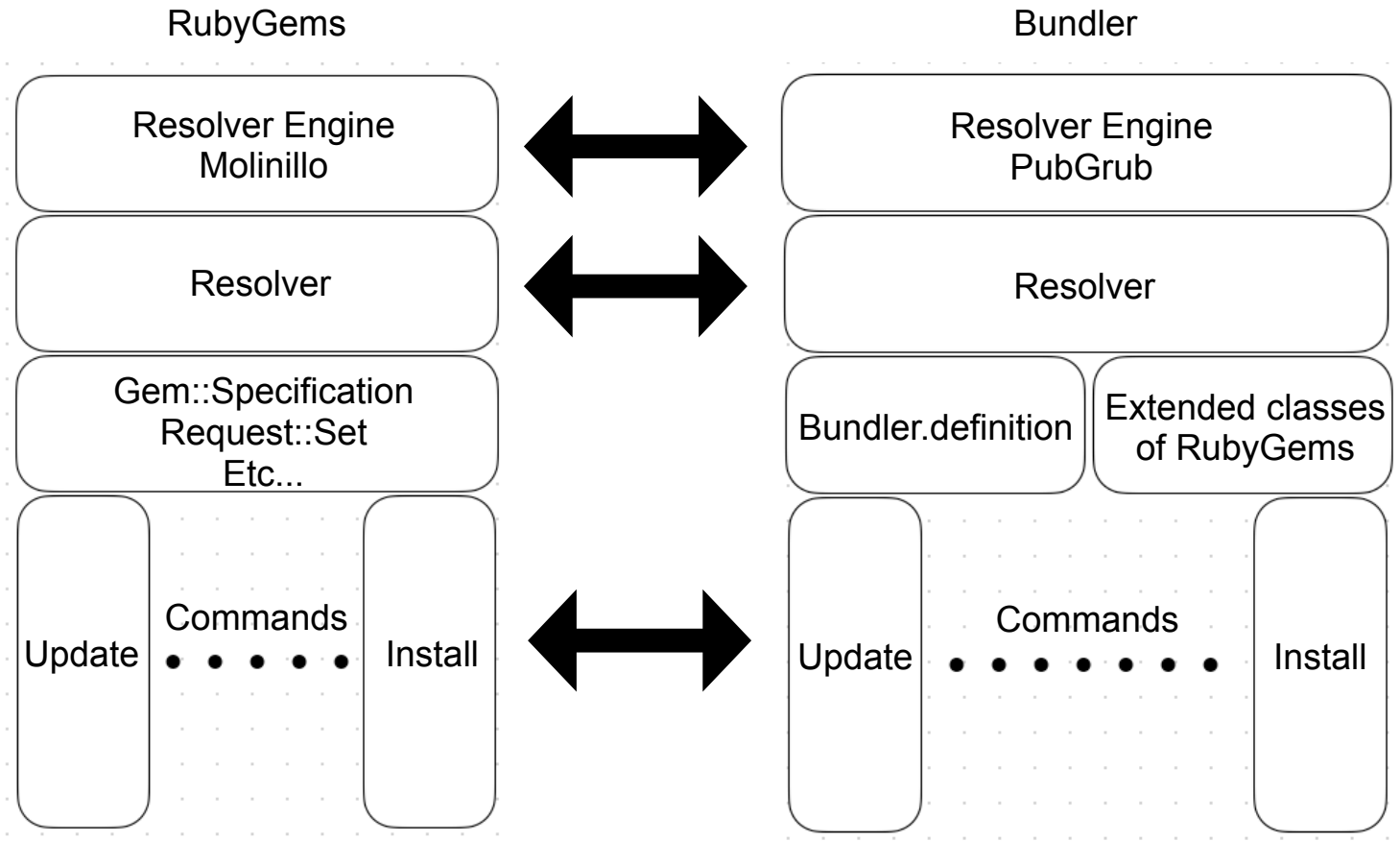
```
# for gem “rails”, git: “https://github.com/rails/rails"

alias_method :rg_extension_dir, :extension_dir
def extension_dir
  # following instance variable is already used in original method
  # and that is the reason to prefix it with bundler_ and add rubocop exception
  @bundler_extension_dir ||= if source.respond_to?(:extension_dir_name)
    unique_extension_dir = [source.extension_dir_name,
File.basename(full_gem_path)].uniq.join("-")
    File.expand_path(File.join(extensions_dir, unique_extension_dir))
  else
    rg_extension_dir
  end
end
```

Copyright © 2020 Present ANDPAD Inc. This information is confidential and was prepared by ANDPAD Inc. for the use of our client. It is not to be relied on by and 3rd party. Proprietary & Confidential 無断転載・無断複製の禁止

Architecture of RubyGems/Bundler

ANDPAD



Copyright © 2020 Present ANDPAD Inc. This information is confidential and was prepared by ANDPAD Inc. for the use of our client. It is not to be relied on by and 3rd party. Proprietary & Confidential 無断転載・無断複製の禁止

Terminology

ANDPAD

- Resolution
  - Ensuring that the dependency constraint are satisfied the combinatorial constraints of multiple libraries.
- Resolver Engine
  - Performs dependency resolution with library name and version combinations and provides a list of libraries if resolved. RubyGems uses Mollinilo, Bundler uses PubGrub
- Resolver
  - Provides the Resolver Engine with the necessary data abstraction and library dependency resolution
  - Provides list of libraries including their names and versions to be installed.

Copyright © 2020 Present ANDPAD Inc. This information is confidential and was prepared by ANDPAD Inc. for the use of our client. It is not to be relied on by and 3rd party. Proprietary & Confidential 無断転載・無断複製の禁止

ANDPAD

Basic case of Gemfile and bundle exec

Copyright © 2020 Present ANDPAD Inc. This information is confidential and was prepared by ANDPAD Inc. for the use of our client. It is not to be relied on by and 3rd party. Proprietary & Confidential 無断転載・無断複製の禁止

Basic case of Gemfile and Bundler

ANDPAD

- How Bundler lock gem versions?
- I'll introduce how resolve these paths with examples.

```
# Gemfile
# frozen_string_literal: true

source "https://rubygems.org"

gem "rss"
```

```
# Gemfile.lock
GEM
  remote: https://rubygems.org/
  specs:
    rexml (3.2.5)
    rss (0.2.9)
      rexml

PLATFORMS
  arm64-darwin-23

DEPENDENCIES
  rss

BUNDLED WITH
   2.5.0.dev
```

Copyright © 2020 Present ANDPAD Inc. This information is confidential and was prepared by ANDPAD Inc. for the use of our client. It is not to be relied on by and 3rd party. Proprietary & Confidential 無断転載・無断複製の禁止

What's Bundler.setup

ANDPAD

- `Bundler.setup` and `Bundler.require` is most important parts of Bundler
  - These methods defined at `runtime.rb`.
- `bundle exec` call `Bundler.setup` and `Kernel.exec`.

```ruby
def setup(*groups)
  @definition.ensure_equivalent_gemfile_and_lockfile if
Bundler.frozen_bundle?

  # Has to happen first
  clean_load_path

  specs = @definition.specs_for(groups)

  SharedHelpers.set_bundle_environment
  Bundler.rubygems.replace_entrypoints(specs)

  # Activate the specs
  load_paths = specs.map do |spec|
    check_for_activated_spec!(spec)

    Bundler.rubygems.mark_loaded(spec)
    spec.load_paths.reject {|path| $LOAD_PATH.include?(path) }
  end.reverse.flatten

  Bundler.rubygems.add_to_load_path(load_paths)

  setup_manpath

  lock(:preserve_unknown_sections => true)

  self
end
```

Copyright © 2020 Present ANDPAD Inc. This information is confidential and was prepared by ANDPAD Inc. for the use of our client. It is not to be relied on by and 3rd party. Proprietary & Confidential 無断転載・無断複製の禁止

Reset your environment with Bundler

ANDPAD

- At first, Bundler update your lockfile and install new versions if it's needed. After that, Reject gem paths that are not `require` yet.

```ruby
@definition.ensure_equivalent_gemfile_and_lockfile if Bundler.frozen_bundle?

# Has to happen first
clean_load_path
```

```ruby
  def clean_load_path
    loaded_gem_paths = Bundler.rubygems.loaded_gem_paths

    $LOAD_PATH.reject! do |p|
      resolved_path = resolve_path(p)
      next if $LOADED_FEATURES.any? {|lf| lf.start_with?(resolved_path) }
      loaded_gem_paths.delete(p)
    end
    $LOAD_PATH.uniq!
  end
```

Copyright © 2020 Present ANDPAD Inc. This information is confidential and was prepared by ANDPAD Inc. for the use of our client. It is not to be relied on by and 3rd party. Proprietary & Confidential 無断転載・無断複製の禁止

How select dependencies by Bundler.definition

ANDPAD

- `Bundler.definition` and `Bundler::Resolver` is core parts for this.
  - Bundler.definition create instance of `Bundler::Resolver` and call resolution methods inside `specs_for`.

```
specs = @definition.specs_for(groups)
SharedHelpers.set_bundle_environment
Bundler.rubygems.replace_entrypoints(specs)
```

- `specs` is instance of Bundler::SpecSet.
  - Bundler inject `bundler` as dependency into Gemfile.

```
>> specs.map(&:name)
=> ["bundler", "rexml", "rss"]
```

Copyright © 2020 Present ANDPAD Inc. This information is confidential and was prepared by ANDPAD Inc. for the use of our client. It is not to be relied on by and 3rd party. Proprietary & Confidential 無断転載・無断複製の禁止

Basic scenario of Bundler resolution

ANDPAD

- `Bundler.rubygems.replace_entrypoints` inject gemspecs of default gems into dependencies of Gemfile.

```
Bundler.rubygems.default_stubs.each do |stub|
  default_spec = stub.to_spec
  default_spec_name = default_spec.name
  next if specs_by_name.key?(default_spec_name)

  specs << default_spec
  specs_by_name[default_spec_name] = default_spec
end
```

specs = Declared in Gemfile: rails, nokogiri, sidekiq, etc...

+

default_spec = Default gems: csv, psych, json, etc...

Copyright © 2020 Present ANDPAD Inc. This information is confidential and was prepared by ANDPAD Inc. for the use of our client. It is not to be relied on by and 3rd party. Proprietary & Confidential 無断転載・無断複製の禁止

Update your LOAD_PATH with scratch

ANDPAD

```
  # Activate the specs
  load_paths = specs.map do |spec|
    check_for_activated_spec!(spec)

    Bundler.rubygems.mark_loaded(spec)
    spec.load_paths.reject {|path| $LOAD_PATH.include?(path) }
  end.reverse.flatten

  Bundler.rubygems.add_to_load_path(load_paths)
  setup_manpath
  lock(:preserve_unknown_sections => true)
  self
```

- These logic is easy to understand. We generate paths generated by resolved gemspec.

  `Gem::Specification#load_paths` returns load paths from gemspec.

- `load_paths` returns like this:

```
~/.rbenv/versions/master/lib/ruby/gems/3.3.0+0/gems/rss-0.2.9/lib
~/.rbenv/versions/master/lib/ruby/gems/3.3.0+0/gems/rexml-3.2.5/lib
~/.rbenv/versions/master/lib/ruby/gems/3.3.0+0/gems/bundler-2.5.0.dev/lib
```

Copyright © 2020 Present ANDPAD Inc. This information is confidential and was prepared by ANDPAD Inc. for the use of our client. It is not to be relied on by and 3rd party. Proprietary & Confidential 無断転載・無断複製の禁止

ANDPAD

How resolve library dependency by Bundler?

Copyright © 2020 Present ANDPAD Inc. This information is confidential and was prepared by ANDPAD Inc. for the use of our client. It is not to be relied on by and 3rd party.　Proprietary & Confidential　無断転載・無断複製の禁止

ANDPAD

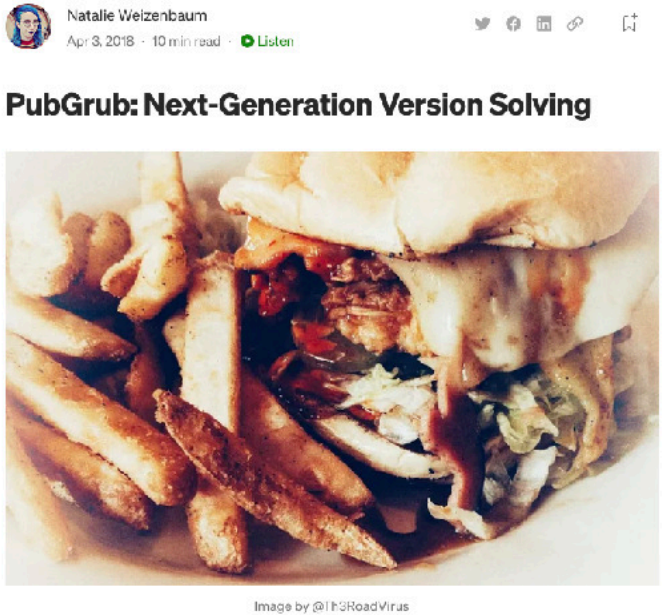
How works PubGrub and Bundler

Copyright © 2020 Present ANDPAD Inc. This information is confidential and was prepared by ANDPAD Inc. for the use of our client. It is not to be relied on by and 3rd party. Proprietary & Confidential 無断転載・無断複製の禁止

What's PubGrub?

ANDPAD

- PubGrub is next generation resolution engine developed by Natalie Weizenbaum a.k.a @nex3.
  - PubGrub is for Dart language. But we have Ruby implementation that is `pub_grub`.
- If resolution conflict occurs with PubGrub, PubGrub give up immediately to resolving loop. This makes faster resolution with complex Gemfile.



Natalie Weizenbaum
Apr 3, 2018 · 10 min read · Listen

PubGrub: Next-Generation Version Solving

Image by @TheRoadVirus

If you'd peeked in on me at work five months ago you'd find me with my nose in a textbook reading up on cutting-edge academic research on constraint-solving problems. Three months ago I was applying that knowledge to code, drafting and tweaking a new algorithm that united that research with a real-world tool. One month ago I was adding the last few features and polishing it up.

https://nex3.medium.com/pubgrub-2fb6470504f

Copyright © 2020 Present ANDPAD Inc. This information is confidential and was prepared by ANDPAD Inc. for the use of our client. It is not to be relied on by and 3rd party. Proprietary & Confidential 無断転載・無断複製の禁止

Easy scenario of PubGrub

ANDPAD

```ruby
source = PubGrub::StaticPackageSource.new do |s|
  s.add 'foo', '2.0.0', deps: { 'bar' => '1.0.0' }
  s.add 'foo', '1.0.0'

  s.add 'bar', '1.0.0', deps: { 'foo' => '1.0.0' }

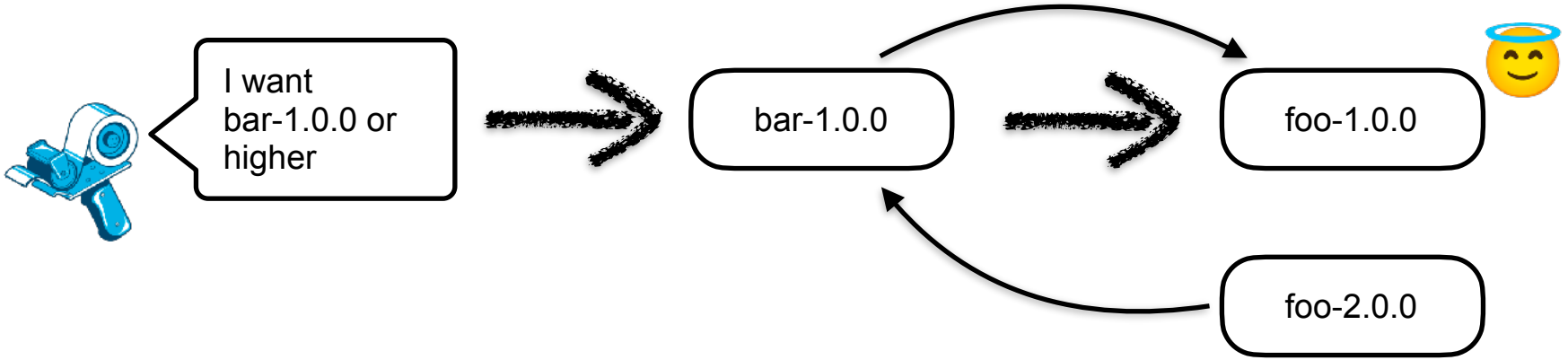
  s.root deps: { 'bar' => '>= 1.0.0' }
end

solver = PubGrub::VersionSolver.new(source: source)
result = solver.solve
p result
#=> {#<PubGrub::Package :root>=>0, "bar"=>#<Gem::Version "1.0.0">,
"foo"=>#<Gem::Version "1.0.0">}
```

- This is basic scenario of dependency resolution.
  - We can see Resolution with PubGrub::VersionSolver and package source definition provided by PubGrub.

Copyright © 2020 Present ANDPAD Inc. This information is confidential and was prepared by ANDPAD Inc. for the use of our client. It is not to be relied on by and 3rd party. Proprietary & Confidential 無断転載・無断複製の禁止

Easy scenario of PubGrub



- We want to use `bar >= 1.0.0`. bar-1.0.0 wants foo-1.0.0.
- We can get resolution result that is `bar-1.0.0` and `foo-1.0.0`.

Copyright © 2020 Present ANDPAD Inc. This information is confidential and was prepared by ANDPAD Inc. for the use of our client. It is not to be relied on by and 3rd party. Proprietary & Confidential 無断転載・無断複製の禁止

Conflict scenario of PubGrub

ANDPAD

```ruby
source = PubGrub::StaticPackageSource.new do |s|
  s.add 'foo', '2.0.0', deps: { 'bar' => '1.0.0' }
  s.add 'foo', '1.0.0'

  s.add 'bar', '1.0.0', deps: { 'foo' => '1.0.0' }

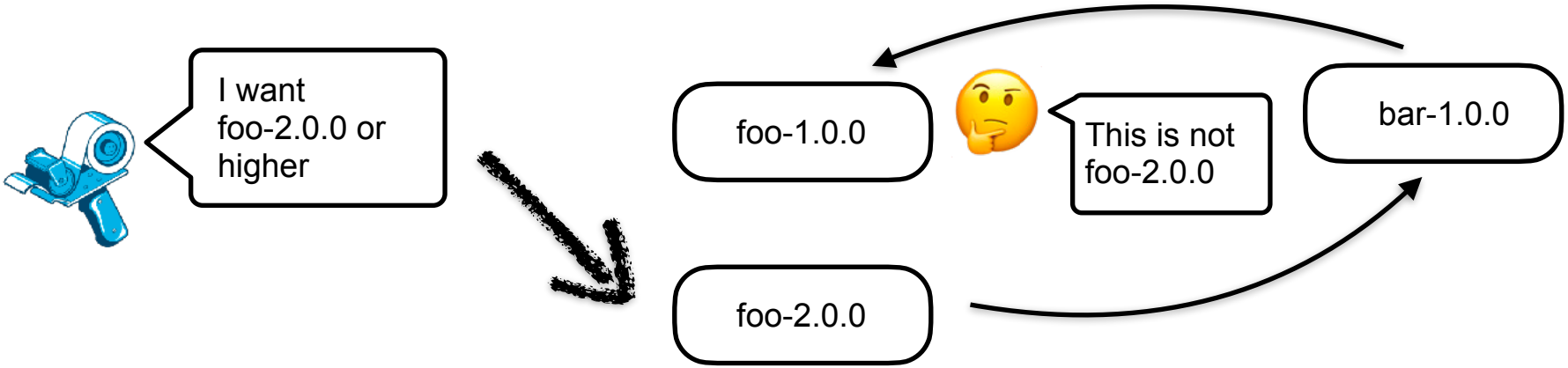
  s.root deps: { 'foo' => '>= 2.0.0' }
end

solver = PubGrub::VersionSolver.new(source: source)
result = solver.solve
p result
#=> pub_grub/version_solver.rb:233:in `resolve_conflict': Could not find compatible
versions (PubGrub::SolveFailure)
```

- This is conflict scenario of dependency resolution.
  - If PubGrub couldn't resolve their versions, it raises `SolveFailure`.

Copyright © 2020 Present ANDPAD Inc. This information is confidential and was prepared by ANDPAD Inc. for the use of our client. It is not to be relied on by and 3rd party. Proprietary & Confidential 無断転載・無断複製の禁止

Easy scenario of PubGrub



- We want to use `foo >= 2.0.0`.
  - But foo-2.0.0 wants bar-1.0.0, and bar-1.0.0 wants foo-1.0.0.

Copyright © 2020 Present ANDPAD Inc. This information is confidential and was prepared by ANDPAD Inc. for the use of our client. It is not to be relied on by and 3rd party. Proprietary & Confidential 無断転載・無断複製の禁止

A bit of complex scenario of PubGrub

ANDPAD

```
source = PubGrub::StaticPackageSource.new do |s|
  s.add 'foo', '3.0.0', deps: { 'bar' => '> 1.0.0' }
  s.add 'foo', '2.0.0', deps: { 'bar' => '1.0.0' }
  s.add 'foo', '1.0.0'

  s.add 'bar', '1.0.0', deps: { 'foo' => '1.0.0' }
  s.add 'bar', '2.0.0'

  s.add 'buzz', '1.0.0', deps: { 'foo' => '> 1.0.0' }

  s.root deps: { 'buzz' => '1.0.0' }
end

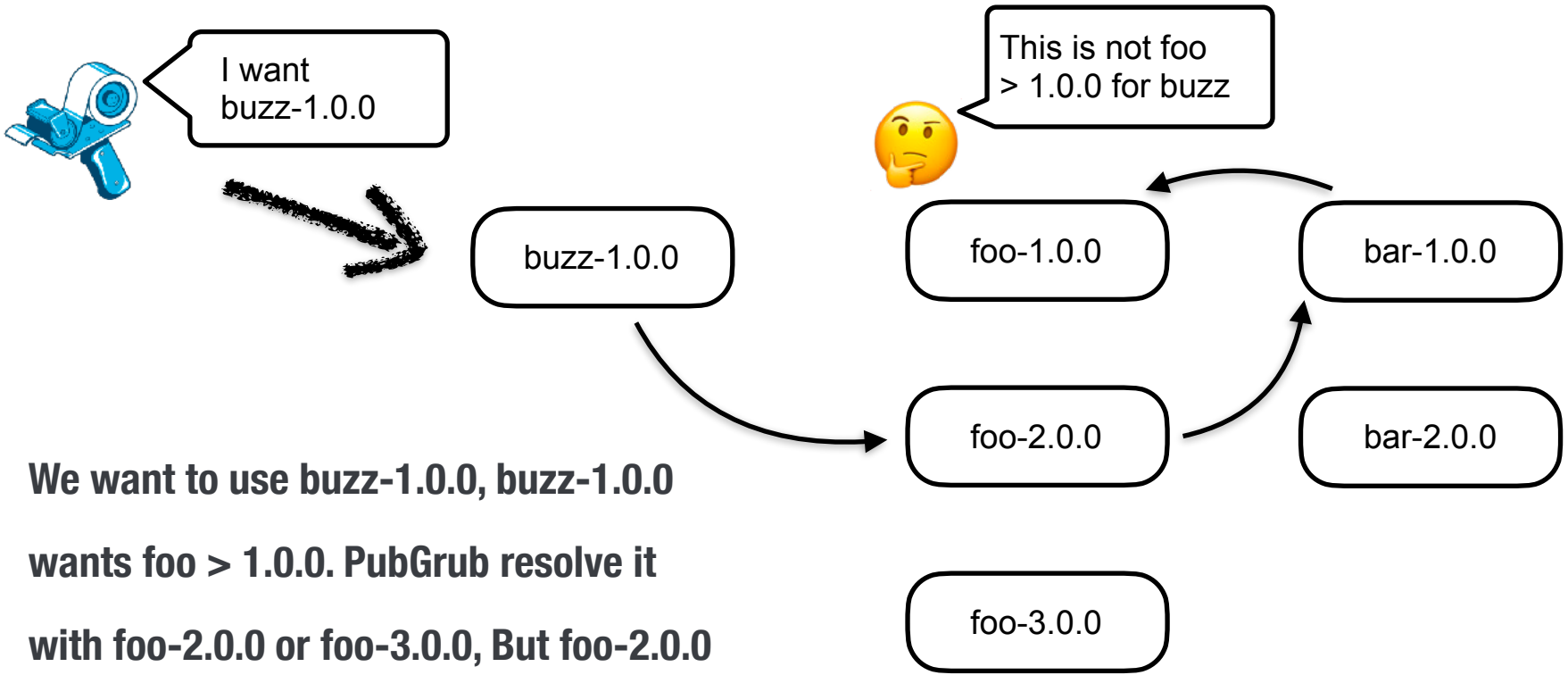
solver = PubGrub::VersionSolver.new(source: source)
result = solver.solve
p result
#=> {#<PubGrub::Package :root>=>0, "buzz"=>#<Gem::Version "1.0.0">, "foo"=>#<Gem::Version
"3.0.0">, "bar"=>#<Gem::Version "2.0.0">}
```

- This is additional scenario for PubGrub. We have three versions of foo, two versions of bar, and buzz.

Copyright © 2020 Present ANDPAD Inc. This information is confidential and was prepared by ANDPAD Inc. for the use of our client. It is not to be relied on by and 3rd party. Proprietary & Confidential 無断転載・無断複製の禁止

A bit of complex scenario of PubGrub

ANDPAD

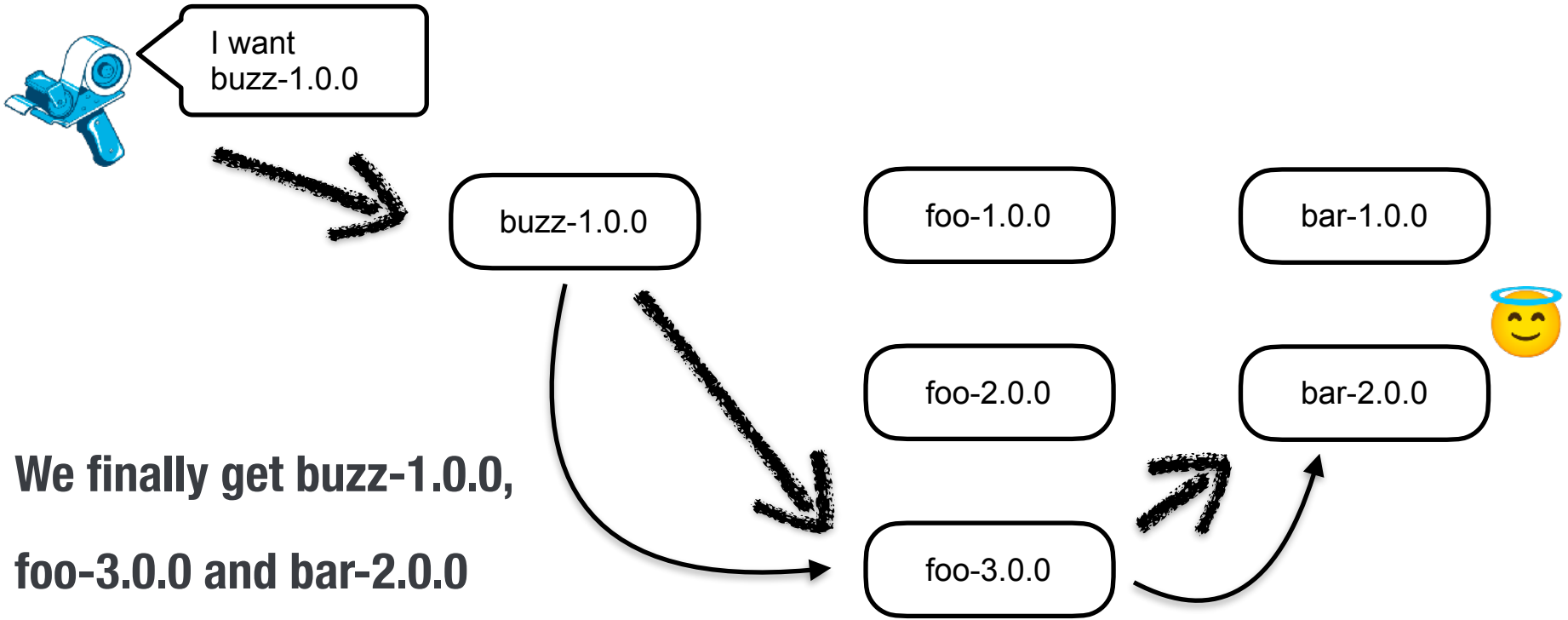


We want to use buzz-1.0.0, buzz-1.0.0 wants foo > 1.0.0. PubGrub resolve it with foo-2.0.0 or foo-3.0.0, But foo-2.0.0 conflicts with bar-1.0.0.

Copyright © 2020 Present ANDPAD Inc. This information is confidential and was prepared by ANDPAD Inc. for the use of our client. It is not to be relied on by and 3rd party. Proprietary & Confidential 無断転載・無断複製の禁止

A bit of complex scenario of PubGrub

ANDPAD



We finally get buzz-1.0.0, foo-3.0.0 and bar-2.0.0 as resolution result.

Copyright © 2020 Present ANDPAD Inc. This information is confidential and was prepared by ANDPAD Inc. for the use of our client. It is not to be relied on by and 3rd party. Proprietary & Confidential 無断転載・無断複製の禁止

ANDPAD

What happened with `bundle update rails`

Copyright © 2020 Present ANDPAD Inc. This information is confidential and was prepared by ANDPAD Inc. for the use of our client. It is not to be relied on by and 3rd party. Proprietary & Confidential 無断転載・無断複製の禁止

Infinitely case of `bundle update`

ANDPAD

```
$ bundle update
Fetching gem metadata from https://rubygems.org/.....
Resolving
dependencies.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....^C
```

Copyright © 2020 Present ANDPAD Inc. This information is confidential and was prepared by ANDPAD Inc. for the use of our client. It is not to be relied on by and 3rd party. Proprietary & Confidential 無断転載・無断複製の禁止

This behavior is derivation of the following events frequently after running `bundle update` at Bundler 2.3 or before.

```
Bundler could not find compatible versions for gem "activesupport":
  In Gemfile:
    inherited_resources (= 1.6.0) was resolved to 1.6.0, which depends on
      has_scope (~> 0.6.0.rc) was resolved to 0.6.0, which depends on
        activesupport (>= 3.2, < 5)

    rails (= 4.2.0) was resolved to 4.2.0, which depends on
      activesupport (= 4.2.0)

Bundler could not find compatible versions for gem "railties":
  In Gemfile:
    inherited_resources (= 1.6.0) was resolved to 1.6.0, which depends on
      railties (>= 3.2, < 5)

    rails (= 4.2.0) was resolved to 4.2.0, which depends on
      railties (= 4.2.0)

    inherited_resources (= 1.6.0) was resolved to 1.6.0, which depends on
      responders was resolved to 1.1.2, which depends on
        railties (>= 3.2, < 4.2)
```

Copyright © 2020 Present ANDPAD Inc. This information is confidential and was prepared by ANDPAD Inc. for the use of our client. It is not to be relied on by and 3rd party. Proprietary & Confidential 無断転載・無断複製の禁止

What's happend called with setup_solver in Bundler?

ANDPAD

- `bundle update` will create instance of `Resolver` for resolution.
- Resolver invoke `setup_solver` and `solve_versions`. `setup_solver` prepared all versions of gemspec(called all_specs) and dependency tree and logger for `solve_versions`

```
source "https://rubygems.org"

gem "rails"
gem "importmap-rails"
```

```
>> @all_specs.keys
=> ["rails", "importmap-rails", "Ruby\u0000", "RubyGems\u0000"]
```

```
>> @all_specs["rails"].map{|s| [s.name, s.version.to_s]}
=>
[["rails", "0.8.0"],
 ["rails", "0.8.5"],
 ...
 ["rails", "7.0.7.2"],
 ["rails", "7.0.8"]]
```

Copyright © 2020 Present ANDPAD Inc. This information is confidential and was prepared by ANDPAD Inc. for the use of our client. It is not to be relied on by and 3rd party. Proprietary & Confidential 無断転載・無断複製の禁止

How use PubGrub in Bundler?

ANDPAD

```
def solve_versions(root:, logger:)
  solver = PubGrub::VersionSolver.new(:source => self, :root => root, :logger => logger)
  result = solver.solve
  result.map {|package, version| version.to_specs(package) }.flatten.uniq
```

- But real case happens resolution conflicts when a dependent gem under rails, such as `railties`, is version-locked by referencing another gem.
  - `PubGrub::SolveFailure` exception occurs and this gem is sent to the retry list.
- Bundler will resolve dependencies defined at Gemfile and all_specs of gem by PubGrub like sample case.

Copyright © 2020 Present ANDPAD Inc. This information is confidential and was prepared by ANDPAD Inc. for the use of our client. It is not to be relied on by and 3rd party. Proprietary & Confidential 無断転載・無断複製の禁止

Real case of PubGrub resolution

ANDPAD



I want
rails-7.0.8
and
importmap-
rails-1.2.1

rails-0.8.0

rails-7.0.8

importmap-rails-0.1.0

importmap-rails-1.2.1

activerecord-...

activemailer-...

activesupport-...

actionview-...

railties-...

actionpack-...

mini_mime-...

mail-...

minitest-...

tzinfo-...

thor-...

rake-...

Copyright © 2020 Present ANDPAD Inc. This information is confidential and was prepared by ANDPAD Inc. for the use of our client. It is not to be relied on by and 3rd party. Proprietary & Confidential 無断転載・無断複製の禁止

Bundler handles conflict result of PubGrub

ANDPAD

```
  rescue PubGrub::SolveFailure => e
    incompatibility = e.incompatibility

    names_to_unlock, names_to_allow_prereleases_for, extended_explanation =
find_names_to_relax(incompatibility)

    names_to_relax = names_to_unlock + names_to_allow_prereleases_for

    if names_to_relax.any?
      (snip)
      root, logger = setup_solver

      Bundler.ui.debug "Retrying resolution...", true
      retry
    end
```

- For example, importmap-rails depends on `railtie`. `importmap-rails` was sent into retry list.
  - `railtie` and `activesupport` are used often as they are rails plugins, so they are almost always included

Copyright © 2020 Present ANDPAD Inc. This information is confidential and was prepared by ANDPAD Inc. for the use of our client. It is not to be relied on by and 3rd party. Proprietary & Confidential 無断転載・無断複製の禁止

Bundler will propagate conflict result into next resolution

ANDPAD



OK, I skip importmap-rails-1.2.1 and its dependencies.

rails-0.8.0

rails-7.0.8

importmap-rails-0.1.0

importmap-rails-1.2.1

activerecord-...

activemailer-...

activesupport-...

actionview-...

railties-...

actionpack-...

mini_mime-...

mail-...

minitest-...

tzinfo-...

thor-...

rake-...

Copyright © 2020 Present ANDPAD Inc. This information is confidential and was prepared by ANDPAD Inc. for the use of our client. It is not to be relied on by and 3rd party. Proprietary & Confidential 無断転載・無断複製の禁止

Result of resolution with rails and importmap-rails



yay, I got the full list of gems with rails-7.0.8 and importmap-rails-1.2.1

rails-7.0.8

importmap-rails-1.2.1

activerecord-...

activemailer-...

.
.
.

actionview-...

```
GEM
  remote: https://rubygems.org/
  specs:
    actioncable (7.0.8)
      actionpack (= 7.0.8)
      activesupport (= 7.0.8)
      nio4r (~> 2.0)
      websocket-driver (>= 0.6.1)
    actionmailbox (7.0.8)
      actionpack (= 7.0.8)
      activejob (= 7.0.8)
      activerecord (= 7.0.8)
      activestorage (= 7.0.8)
      activesupport (= 7.0.8)
      mail (>= 2.7.1)
      net-imap
      net-pop
      net-smtp
    actionmailer (7.0.8)
      actionpack (= 7.0.8)
      actionview (= 7.0.8)
      activejob (= 7.0.8)
      activesupport (= 7.0.8)
      mail (~> 2.5, >= 2.5.4)
```

Copyright © 2020 Present ANDPAD Inc. This information is confidential and was prepared by ANDPAD Inc. for the use of our client. It is not to be relied on by and 3rd party. Proprietary & Confidential 無断転載・無断複製の禁止

ANDPAD

Proposals for the future

Copyright © 2020 Present ANDPAD Inc. This information is confidential and was prepared by ANDPAD Inc. for the use of our client. It is not to be relied on by and 3rd party.　Proprietary & Confidential　無断転載・無断複製の禁止

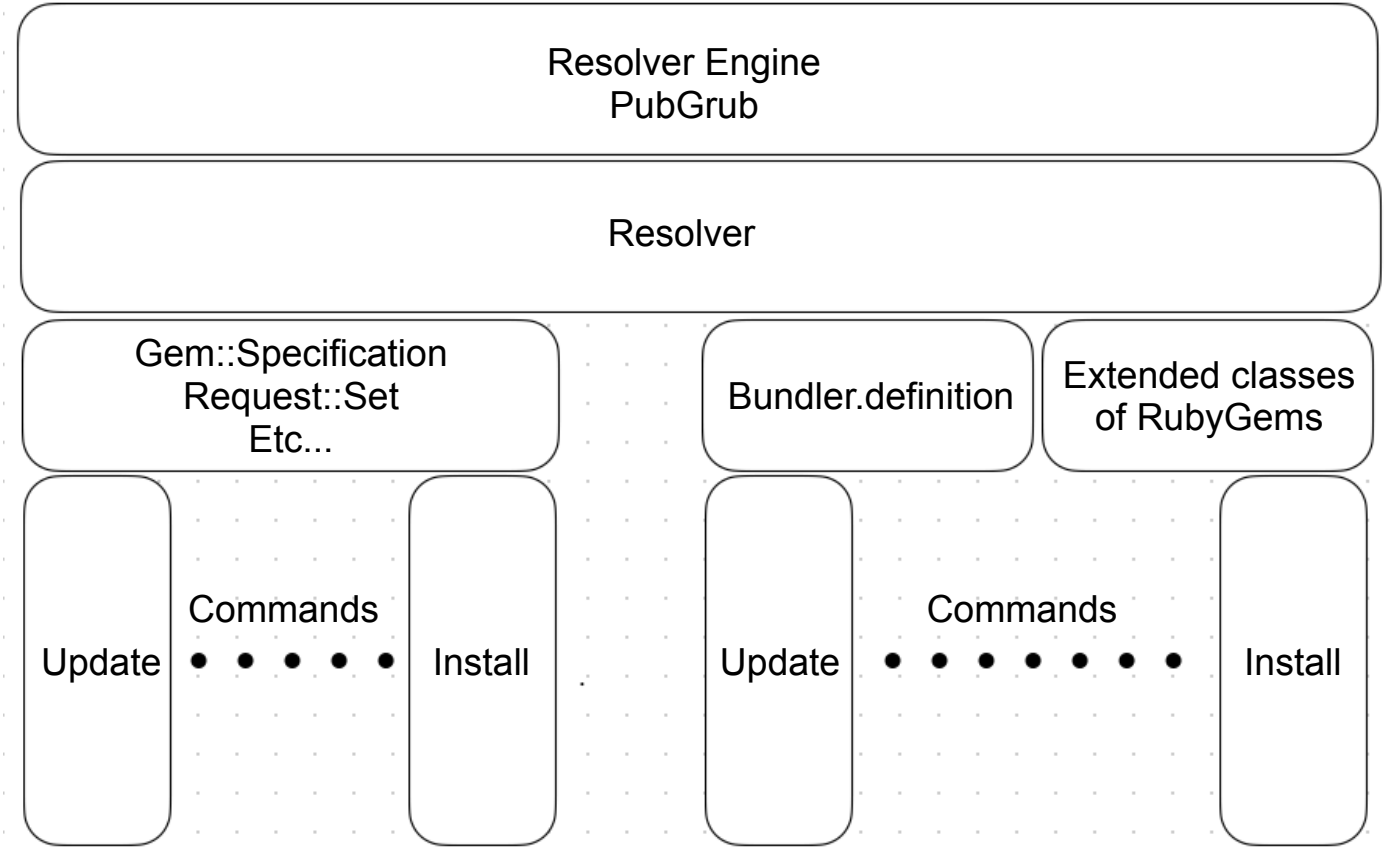
ANDPAD

Resolve duplicates and redundant of code and commands

Copyright © 2020 Present ANDPAD Inc. This information is confidential and was prepared by ANDPAD Inc. for the use of our client. It is not to be relied on by and 3rd party. Proprietary & Confidential 無断転載・無断複製の禁止

Migration RubyGems and Bundler in the future

ANDPAD



Copyright © 2020 Present ANDPAD Inc. This information is confidential and was prepared by ANDPAD Inc. for the use of our client. It is not to be relied on by and 3rd party. Proprietary & Confidential 無断転載・無断複製の禁止

ANDPAD

- I talked about...
  - Knowledge RubyGems, Bundler and Package Manager.
  - How works Bundler modify LOAD_PATH of Ruby
  - How works PubGrub and Bundler



< Ruby is a programmer's best friend

Copyright © 2020 Present ANDPAD Inc. This information is confidential and was prepared by ANDPAD Inc. for the use of our client. It is not to be relied on by and 3rd party. Proprietary & Confidential 無断転載・無断複製の禁止